

Gestão e Histórico de Releases

Código do Documento: GES-HTR-REL-001	Versão: 1.0
Data de Emissão: Janeiro de 2026	Classificação de Informação: Uso Interno / Clientes / Auditoria
Área Responsável: Engenharia de Software / DevOps	Aplica-se a: Todos os Serviços e Microserviços

1. OBJETIVO

Esta declaração formal visa comprovar a existência e a aplicação contínua de processos padronizados de gestão de releases, controle de versão e auditoria de alterações no código-fonte dos sistemas e serviços mantidos pela organização.

2. ESTRUTURA E REPOSITÓRIO DE CÓDIGO (GITHUB)

A gestão de versões de todos os serviços da organização é centralizada no provedor **GitHub**, utilizando repositórios privados e seguros. O controle de mudanças e histórico de lançamentos é rastreado de forma automatizada atrelado à esteira de integração e entrega contínuas (CI/CD).

2.1. Modelo de Ramificação (Branching Strategy)

Os repositórios seguem um fluxo de trabalho estruturado, com ramificações (*branches*) específicas destinadas à segregação de ambientes e garantia de qualidade:

Branch / Ramo	Ambiente Associado	Finalidade e Critérios de Aprovação
<code>main</code> / <code>master</code> / <code>prod</code>	Produção (Prod)	Código em execução operacional. Alterações entram via Pull Requests (PRs) aprovados após validação em homologação e testes.
<code>staging</code> / <code>stage</code>	Homologação (Stage)	Ambiente pré-produção utilizado para validação final e aprovação do time de produto/cliente.
<code>qa</code> / <code>testing</code>	Qualidade / Testes (QA)	Ambiente dedicado para serviços aplicáveis, onde são executados testes de regressão e garantia de qualidade antes do merge para staging.

<code>feature/*</code> ou <code>fix/*</code>	Desenvolvimento Local	Ramos isolados para implementação de melhorias ou correções específicas, originando Pull Requests para revisão de código.
--	-----------------------	---

3. MEIOS DE REGISTRO E RASTREABILIDADE DE HISTÓRICO

O histórico completo de revisões, modificações relevantes e pacotes liberados em produção é mantido através das seguintes ferramentas nativas do GitHub:

- GitHub Releases & Tags: As versões oficiais enviadas para produção são marcadas por meio de *Tags* e consolidadas na aba *Releases* do repositório, adotando versionamento semântico (ex.: `v2.5.0`).
- Commits Auditáveis: Cada commit contém autor, data, hora e mensagem descritiva da alteração realizada no sistema.
- Rastreabilidade por Pull Requests (PRs): Todo envio de código para as branches de `qa`, `stage` e `prod` exige abertura de PR, vinculando a alteração à tarefa correspondente.
- GitHub Actions / CI-CD Logs: Cada alteração aprovada aciona pipelines automatizadas cujos históricos de compilação, testes e *deploy* permanecem salvos e auditáveis.